CS 261 - Security in Computer Systems
Lecture: Network Security

Scribed by: Aisha Mushtaq

6 September 2017

The topic for the class was 'Network Security' and the The assigned reading for this class, 'A Look Back at Security Problems in the TCP/IP Protocol Suite' (published in 2004), seemed rather dated and dubious in its applicability today.

1 Threat models

Let's start by describing potential threat models. A threat model basically defines what's in scope, what's not in scope and what the capabilities of the attacker are. It is used to evaluate the security of a system. There are two views of the world when we look at threat models for network security:

1. Cryptographers' view
2. Network security view.

1.1 Cryptographers' view

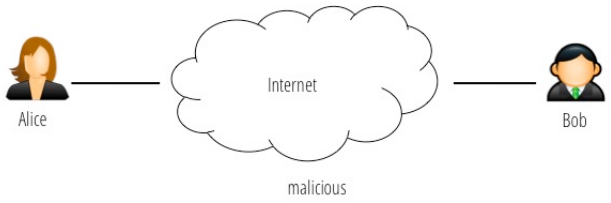

Figure 1: Cryptographer's view of the world

The cryptographer's view, shown in figure 1, assumes that Alice is at one edge, Bob is at the other edge and the cloud in between them is the Internet. Anything bad can happen in the Internet, for instance, the attackers can read data, corrupt the data, etc. The solution for this view mainly involves encrypting the data such that no malicious activity can happen in the insecure cloud.

1.2 Network security view

The network security view assumes Alice and Bob are at the edges, and are connected by a bunch of links and routers, as shown in figure 2. So let's try to envision **'what a bad guy could do?'** in this model. If the adversary controls one of them how much harm can they do?

$\implies$ The answer is: it depends on which path/node they control.

At this point, the adversaries bifurcate into two categories.

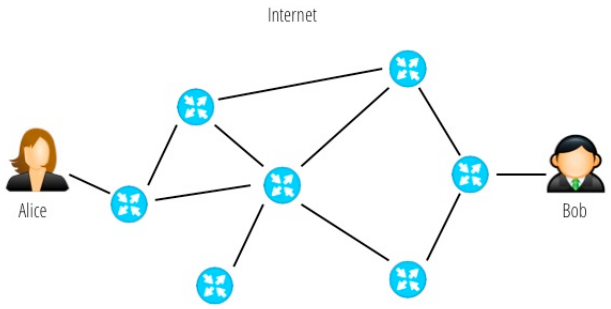

Figure 2: Network security's view of the world

1. On-path adversary : An adversary that lies on the path of communication between Alice and Bob. This type of adversary is more dangerous, as he/she potentially has a lot of control.
2. Off-path adversary : An adversary that does not lie on the direct path of communication between Alice and Bob. Seemingly, they are less dangerous, but there are potentially many more of them to be aware of. One example of an off-path attack: Denial of Service attacks (adversary just throws a lot of packets at you).

Why use the network security model?

The network security model is cheaper to defend, as you can address more specific issues instead of viewing the problem in an abstract sense. It also potentially saves us from the problem of securely distributing encryption keys.

2 Case study: Wifi at Starbucks

We look at a local attacker scenario in the following case study. Imagine Alice connects to a wifi access point (AP) in Starbucks and the attacker, Mallory, also connects to the same AP as shown in figure 3.

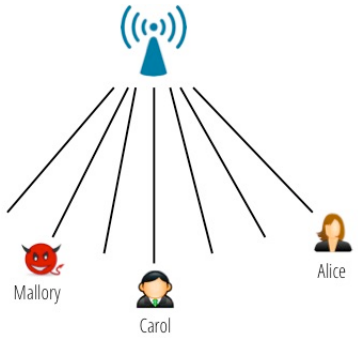

Figure 3: Starbucks Case Study

In this particular scenario, what malicious behavior can Mallory engage in?

- Eavesdrop and listen to Alice's packets.
- Send false packets to Alice

- Re-write packets someone else sent to Alice. Can Mallory edit packets? It is much harder to edit packets during transmission. How can she achieve this? listen[1] and jam the tail-end of the transmission using interference. Then, she can re=transmit the packet of her choice. Another way to edit the packet would be to transmit her doctored packet with the same TCP sequence number as the intercepted packet. The TCP protocol de-duplicates packets, and depending on Alice's OS, the newer or older packet may be kept.

Does the AP filter packets with incorrect an source IP address (i.e. if someone local is pretending to be google.com)?

Currently, the APs do not bother checking IP address authenticity, they just look at MAC addresses.
Even if APs were to check IP addresses, Mallory could pretend to be someone local, such as Carol from figure 3. Alternatively, Mallory could also broadcast radio waves with fake source IP addresses.

2.1 DHCP (Dynamic Host Configuration Protocol)

A host uses DHCP to discover its own IP address, the IP address for its local DNS name server and IP address for its first-hop gateway router.
When Alice connects to the network, she needs to boot up; i.e., she needs to figure out the IP address assigned to her machine, gateway router, DNS server etc. To do this, she broadcasts a DHCP request to get set up. Everyone else knows to ignores the message except the DHCP server, which responds to her request by assigning her machine an IP address and telling it how to reach the gateway router.

How can Mallory use this to her advantage?

Mallory can respond to Alice's DHCP request before the DHCP server does, and declare herself as the gateway router. Mallory is now on-path, she thus has increased her attack potential considerably.

What if there is an IP address conflict, when Mallory assigns Alice an IP address?

To make sure the attack succeeds, Mallory can send a legitimate IP address which she can reserve in advance from the DHCP server.

2.2 Ethernet instead of Wifi

Ethernet is a shared medium like wifi, using a physical wire instead of radio waves. Mallory can carry out a similar attack discussed above using ARP spoofing.

For all the scenarios discussed above, Mallory can potentially have many more attack vectors with higher level protocols.

3 Case study: BGP routing

We start by describing a simplified view of how BGP routing works. In our network security model above, we had a bunch of nodes connected by links. Each of these links have an associated cost[2]. Based on the cost, the nodes have routing tables which maintain the next hop and the best cost (depends on multiple factors) to destinations. No router maintains global knowledge. Instead, each router just remembers the next hop and the total cost. The router periodically broadcasts the information it possesses to its neighbors and the nodes/routers remember the best one (while trusting that the knowledge it received is correct). These periodic updates are propagated to all routers, which eventually leads to the network's convergence.

[1] She would require fast hardware to do this, regular wifi cards would not work
[2] This cost is a monetary cost, not the hop-count / shortest-path

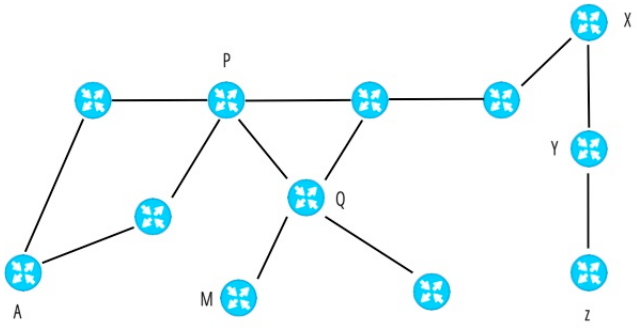

Figure 4: BGP routing

To avoid loops, AS-paths (the list of Autonomous Systems that the packet will traverse from the current location to the destination) are also announced with the next hop.

Attacks on BGP

In figure 4, assume host A is sending packets to host Z.
In the trivial case, if Mallory (our attacker, labeled as M in 4) is on-path, she can easily compromise A and Z's communication. However, if Mallory is off-path and wants to compromise A and Z's communication, what can she do?

Mallory could announce a fake lower-cost path to A in order to reach Z. This could result in the following:

- A's packets could potentially not reach Z, thus compromising availability
- If A's data is not encrypted, M could compromise confidentiality as well.
- Mallory could potentially set up a wormhole to Z as in 5. She could send the packets through this wormhole to Z after manipulating the packet's contents.

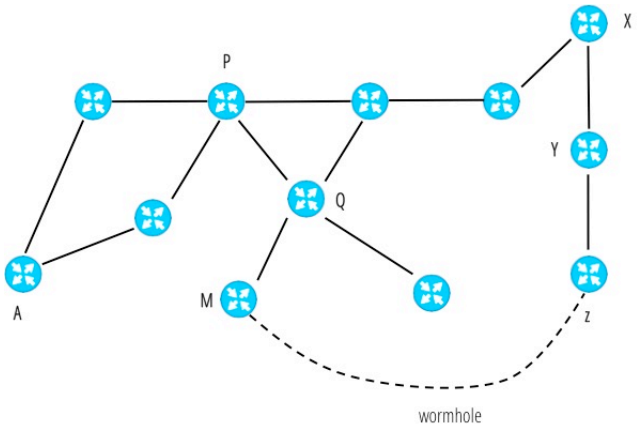

Figure 5: BGP routing with Mallory's wormhole

How could Mallory achieve this wormhole?

Mallory could use the AS-path feature of BGP to trick the protocol. She would pick a path she knows to Z and include that in the fake low-cost route announcements. This would pollute all paths, except the path that she included in the announcement. Essentially, Mallory has set up a private path to Z now for which she is on-path.

Examples of such attacks:

- Pakistan announced a lower cost fake path and took down YouTube.
- An attacker spoofed the IP address of a central server of a bitcoin mining pool in order to get their proof-of-works, but never paid them.
- Due to a BGP misconfiguration where a shorter path was accidentally announced, intra-city communication for Denver, Colorado passed through Iceland for 15 minutes, considerably slowing down the city's internet.

3.1 Potential Defenses

Issue: Mallory controls one router, and all the routers believe other routers to be trustworthy, which potentially pollutes the entire network.

Solution: One half-measure is to have multiple routes announced, so at least some of the packets get through to Z. The better solution is as follows. We assume everyone knows Z's public key. Z should sign the route announced by M to Z using it's private key. Everyone can verify this signature using Z's public key. Routers only accept the route if it is signed by Z, or by a chain that connects to Z, and not otherwise. This will lead to a chain of signed routes as shown in figure 6.

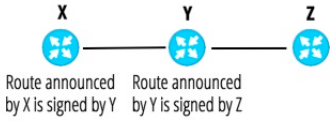

Figure 6: Chain of signed routes

One caveat of this solution is that everyone needs to know the public keys of all the routers.

4 Great Firewall of China and Great Cannon of China

For the student presentation we mainly discussed two things:

- The Great Firewall of China
- The Great Cannon of China

4.1 The Great Firewall of China (GFW)

The GFW is an on-path attack system that eavesdrops on traffic between China and the rest of the world, blocking any requests for banned content.
It has servers lining the border of China (shown in figure 7), which keep track of connections and reassembles their packets, instead of considering each packet in isolation, to determine if it should block traffic. It terminates the connection by injecting a series of forged TCP Reset (RST) packets that tell both the requester and the destination to stop communicating.

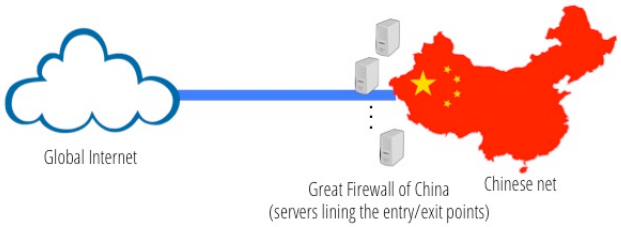

Figure 7: Great Firewall of China

4.2 The Great Cannon (GC)

The Great Cannon (GC) is a censorship tool employed in China to launch a Distributed Denial-of-Service attack on organizations dedicated to resisting the Chinese government's censorship laws (such as GreatFire.org).

The GC system is an in-path adversary, instead of being an on-path adversary, i.e. it is capable of not only injecting traffic but also directly suppressing traffic.

GC targets users accessing Chinese sites from outside China and enlists them to launch a DDOS later. For example, if the user was requesting a Javascript file (like js files for advertisements on websites) from Baidu's server (a Chinese web search site), the GC would drop the request before it reached Baidu and would instead send malicious Javascript back to the requesting user.

Some salient features of the GC include:

- GC ignores 98% of potential targets, and randomly decides on 2% of the requesting users to send the malicious code files to. This makes the attack harder to detect, but is enough to proliferate through.
- The GC is aimed at only specific destination IP addresses.
- GC servers appear to be co-located with the GFC. The authors of the paper varied the TTL field in packet headers to detect the location of GFC's and GC's servers.